



HAL
open science

Kinematical performances in 5-axis machining

Sylvain Lavernhe, Christophe Tournier, Claire Lartigue

► **To cite this version:**

Sylvain Lavernhe, Christophe Tournier, Claire Lartigue. Kinematical performances in 5-axis machining. IDMME, May 2006, Grenoble, France. 12p. hal-00999718

HAL Id: hal-00999718

<https://hal.science/hal-00999718>

Submitted on 3 Jun 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

KINEMATICAL PERFORMANCES IN 5-AXIS MACHINING

Sylvain Lavernhe

ENS de Cachan

Christophe Tournier

ENS de Cachan

Claire Lartigue

IUT de Cachan – Université Paris Sud 11

Laboratoire Universitaire de Recherche en Production Automatisée

ENS de Cachan – Université Paris Sud 11

61 avenue du Président Wilson, 94235 Cachan cedex - France

name@lurpa.ens-cachan.fr

Abstract:

This article presents a predictive model of the kinematical behaviour during 5-axis machining. This model highlights differences between the programmed tool-path and the actual follow-up of the trajectory. Within the High Speed Machining context, kinematical limits of the couple CNC-machine-tool have to be taken into account in the model. The originality of the model is the use of the inverse-time method to coordinate machine-tool axes, whatever their nature (translation or rotation). The model reconstructs the actual relative velocity tool-surface from each axis velocity profile highlighting trajectory portions for which cutting conditions are not respected.

Key words: 5-axis machining, high speed machining, predictive model, kinematical behaviour

1 Introduction

Today, machining of aeronautic free form surfaces (turbine blades...) is carried out using multi-axis milling centre. The use of rotational axes authorizes the choice of the tool orientation relatively to the surface. Hence, these supplementary degrees of freedom allow reduction costs, improving the surface quality and decreasing machining time. In the context of High Speed Machining (HSM), axis and structure solicitations are especially high. The machining process is modified and may alter the surface quality: velocity drops cause marks, vibrations and high solicitations of each axis generate bad cutting conditions.

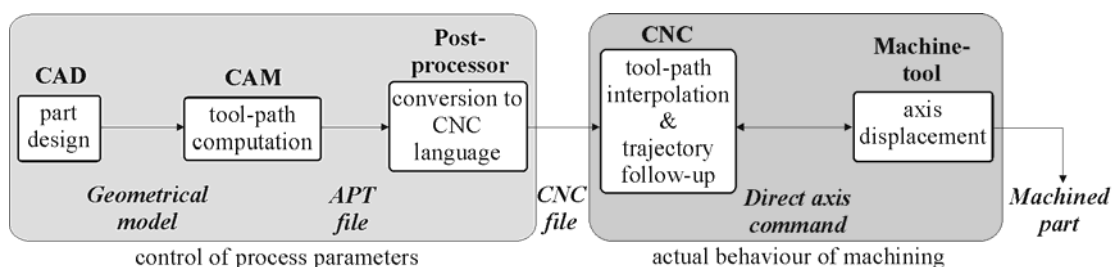


Figure 1. Structure of the process.

As the machined surface results from each stage of the process, numerous parameters have to be taken into account to control the final result (Figure 1). The Computer Aided Machining (CAM) stage computes the tool-path, which constitutes an approximation of the CAD model according to the interpolation format (linear or polynomial), the driving tool direction and the CAM parameters (machining tolerance, scallop height). The choice of the parameter values directly influences the surface quality. Following, the post-processing stage converts the calculated tool-path into an adapted file for the numerical controller (NC), called the CNC file. The CNC file contains the set of tool positions and axis orientations and the corresponding feedrates. In 5-axis machining, the post-processor may also solve the Inverse Kinematical Transformation (IKT) to express the tool-path into direct axis commands. The tool-path interpolation and the trajectory follow-up are performed by the CNC. As this stage, the follow-up strongly depends on the CNC parameters such as time cycles, velocity limitations and special functions (look-ahead). The kinematical behaviour during machining is also affected by the machine-tool architecture and axis capacities. Therefore, from the CAM stage to the actual machining, numerous parameters influence kinematical performances in 5-axis machining which may alter the surface quality and machining time. In particular, differences exist between the calculated trajectory and associated feedrates defining the CNC file and the actual tool movement relative to the surface.

This paper deals with a predictive model of the machining behaviour with the objective of evaluating the actual relative velocity tool-surface during machining from the CNC file. For a couple machine-tool - CNC, the model predicts the kinematical behaviour through axis velocities. The final purpose is to find an optimal strategy which allows respecting programmed feedrate as well as possible. The methodology proposed consists in three main steps (Figure 2). First, the programmed tool-path ($X_{pr}, Y_{pr}, Z_{pr}, i, j, k$) is transformed into a trajectory in the articular space (X_m, Y_m, Z_m, A, C) by solving the IKT. Once axes are coordinated, the post-processor generates the velocity profiles for each axis: velocity limitations along the trajectory resulting from axis and CNC capacities are computed; a model of the CNC treatments and the special functions is established. The originality of this stage is the use of the inverse-time method, allowing a similar treatment for translation axes and rotation axes. Finally, the relative velocity tool-surface is reconstructed considering the machine-tool architecture allowing the prediction of bad cutting condition zones. In a first approach the linear interpolation only is modelled. The application is carried out on a 5-axis milling centre Mikron UCP 710 with an industrial numerical controller Siemens 840D.

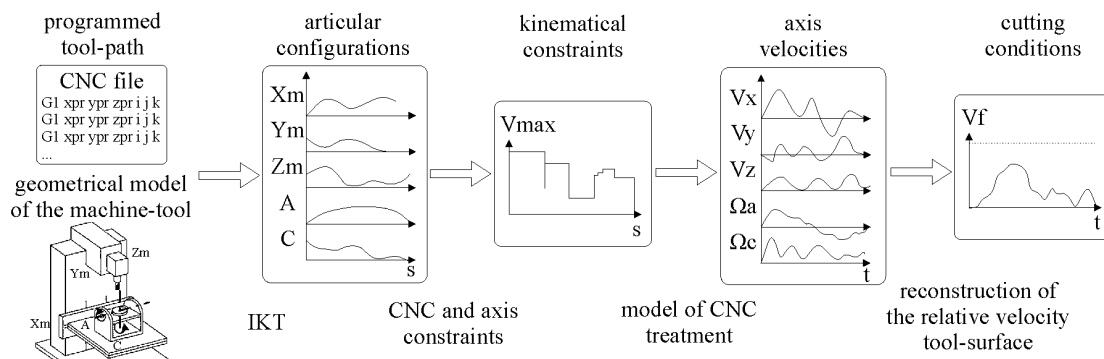


Figure 2. Structure of the predictive model.

The next section deals with the IKT for a CAXYZ serial architecture. It underlines the difficulty in the choice of solution to match the CNC behaviour. Generation of axis

profiles is detailed in section 3. The inverse-time method used to express velocity constraints is exposed. Limitations from CNC and axis capacities are formulated to construct kinematical profiles. An illustration is exposed in section 4. In section 5, the relative velocity tool-surface is recalculated from each axis velocity.

2 Trajectory into the articular space

The tool trajectory results from the axis displacements of the machine-tool. For each Cutter location (Cl point) computed by the CAM software, the tool position and its axis orientation relative to the surface are achieved by combining axis articular positions. From the programmed tool-path, the IKT calculates corresponding articular configurations. Two calculation modes are possible, the IKT can be performed in real time by the NC unit during machining or it can be computed off-line by a dedicated post-processor [1]. As our objective is to predict articular configurations computed in real time by the NC unit, a post-processor is created to simulate the IKT by the CNC. Therefore, the role of this post-processor is to compute the articular configurations (X_m, Y_m, Z_m, A, C) for each Cl point $(X_{pr}, Y_{pr}, Z_{pr}, i, j, k)$ programmed in the part frame (Figure 3).

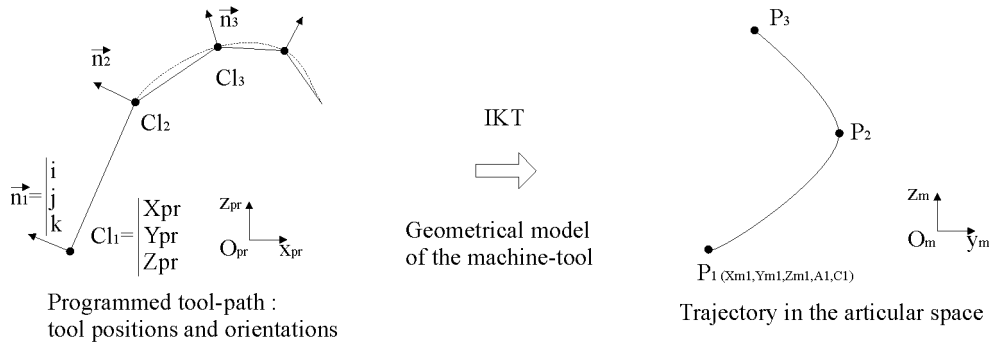


Figure 3. Inverse Kinematical Transformation.

Given the machine-tool architecture (Figure 13), the IKT leads to equations (1) and (2). Details on the geometrical modelling of the machine-tool and the IKT formulation are exposed in appendix.

$$\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}_{(x_m, y_m, z_m)} = P_{mb} \times P_{bp} \times P_{ppr} \times \begin{bmatrix} i \\ j \\ k \\ 0 \end{bmatrix}_{(x_{pr}, y_{pr}, z_{pr})} \quad (1)$$

$$\begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix}_{(O_m, x_m, y_m, z_m)} = P_{mb} \times P_{bp} \times P_{ppr} \times \begin{bmatrix} X_{pr} \\ Y_{pr} \\ Z_{pr} \\ 1 \end{bmatrix}_{(O_{pr}, x_{pr}, y_{pr}, z_{pr})} \quad (2)$$

Equation (2) directly gives the axis commands (X_m, Y_m, Z_m) . If we consider that the axes of the programming basis and the table basis are parallel, equation (1) leads to the following system of equations:

$$\begin{cases} i = \sin(C) \times \sin(A) \\ j = -\cos(C) \times \sin(A) \\ k = \cos(A) \end{cases} \quad (3)$$

System (3) has two domains of solutions ($A_1 > 0$ or $A_2 < 0$). In function of the (i, j, k) values, solutions vary (Table 1).

| | $i < 0$ | $i = 0$ | $i > 0$ |
|---------|---|---------------|--|
| $j < 0$ | $A_1 = \text{acos}(k) \quad C_1 = -\text{atan}(i/j)$ | | |
| | $A_2 = -\text{acos}(k) \quad C_2 = -\text{atan}(i/j) + \pi$ | | |
| $j = 0$ | $A_1 = \text{acos}(k) \quad C_1 = -\pi/2$ | $A = 0$ | $A_1 = \text{acos}(k) \quad C_1 = \pi/2$ |
| | $A_2 = -\text{acos}(k) \quad C_2 = \pi/2$ | C undefined | $A_2 = -\text{acos}(k) \quad C_2 = -\pi/2$ |
| $j > 0$ | $A_1 = \text{acos}(k) \quad C_1 = -\text{atan}(i/j) + \pi$ | | |
| | $A_2 = -\text{acos}(k) \quad C_2 = -\text{atan}(i/j)$ | | |

Table 1. Domains of solutions (A_1, C_1) and (A_2, C_2) .

Due to physical limitations, the range of angle A is limited to $[-30^\circ; +120^\circ]$. So, all solutions are not practically possible. Different possible cases are summarized in Table 2.

| values of k | $[-1; -0.5 [$ | $[-0.5; 0.866 [$ | $[0.866; 1 [$ | 1 |
|---------------------|---------------|------------------|------------------------------|----------------------------------|
| number of solutions | 0 | 1 | 2 | ∞ |
| | no solution | (A_2, C_2) | (A_1, C_1) or (A_2, C_2) | $A=0$ and $C=\text{unspecified}$ |

Table 2. Sets of solutions for the IKT.

When several solutions exist, the choice of one of them influences the axis behaviour. Indeed, switching from domain 1 ($A_1 > 0$) to domain 2 ($A_2 < 0$) makes the table C rotates about 180° . This rapid movement can cause rear gouging, especially when using toroidal tool [2][3]. Moreover, during this movement, the displacement of the cutter contact on the surface is small; the relative velocity tool-surface is almost zero. As cutting conditions are not satisfied, velocity drops will cause marks on the surface. The discussion on the best choice that can be done according to the kinematical behaviour is the subject of current work and will not be discussed here. The objective of the post-processor is to approach the actual behaviour of IKT performed by the CNC. During tests on our system, incoherent movements can be noticed. The choice of solution done in real time may be arbitrary. Indeed, for two similar trajectories, the CNC can choose two different solutions; thus an inversion on the A axis appears whereas it is not necessary. Moreover, for certain cases, chosen solution leads to axis displacements out of range. So, predict what will be the solution chosen by the CNC is difficult. After having tested several cases on the CNC, the post-processor tries to choose the solution that matches the one done by the CNC.

At this stage, axis configurations corresponding to Cl points are computed.

3 Prediction of axis velocities

Prediction of the relative velocity tool-surface requires the estimation of each axis velocity. Axis kinematical behaviour during the follow-up of the trajectory depends on the tool-path geometry and the performance of the components (CNC, machine-tool axis...). Several limitations linked to the HSM context have to be taken into account to generate kinematical profiles.

3.1 Programming method

Usually, in 3-axis machining, axis displacements of the machine-tool correspond to the displacements of the tool relatively to the surface. For multi-axis machining, due to the rotational axes, axis displacements are totally different. During the follow-up, the main difficulty is to coordinate translational and rotational axes. This difficulty is emphasised by the differences between the kinematical capacities of rotational axes and translational axes. The proposed method consists in programming velocities by the inverse-time. It defines axis velocities by specifying the time to go from one point to the following one. As a result it is easier to coordinate axes with different natures of movements (translation, rotation). As many axes as necessary can be interpolated. For instance, when programming the tool-path in the part frame, the feedrate V_f is defined as constant between two Cl points (Cl_1, Cl_2). If the length of the segment is L , the tool is supposed to move during Δt from Cl_1 to Cl_2 (eq. (4)).

$$\Delta t = \frac{L}{V_f} \quad \text{with} \quad L = \|\overrightarrow{Cl_1 Cl_2}\| \quad (4)$$

To each couple Cl_i-n_i programmed in the part frame corresponds one axis configuration P_i (P_i axis 1, P_i axis 2, P_i axis 3 ...). In the articular space, the time allocated to each axis to go from one position $P_{1 \text{ axis } j}$ to the following one $P_{2 \text{ axis } j}$ should be Δt . Thus, for the segment (Cl_1, Cl_2) the axis speeds to apply to the machine-tool are:

$$V_{12 \text{ axis } j} = \Delta P_{12 \text{ axis } j} \times V_{\text{time inverse}} \quad \text{with} \quad \Delta P_{12 \text{ axis } j} = P_{2 \text{ axis } j} - P_{1 \text{ axis } j} \quad \text{and} \quad V_{\text{time inverse}} = \frac{1}{\Delta t} \quad (5)$$

By this way, the calculus of axis velocities is adapted for linear interpolation in the articular space. However, the programmed tool-path is theoretically interpolated in the part frame. The trajectory could be curve in the articular space between two axis configurations. Hence, differences can appear, especially if distances between successive configurations are high (Figure 4).

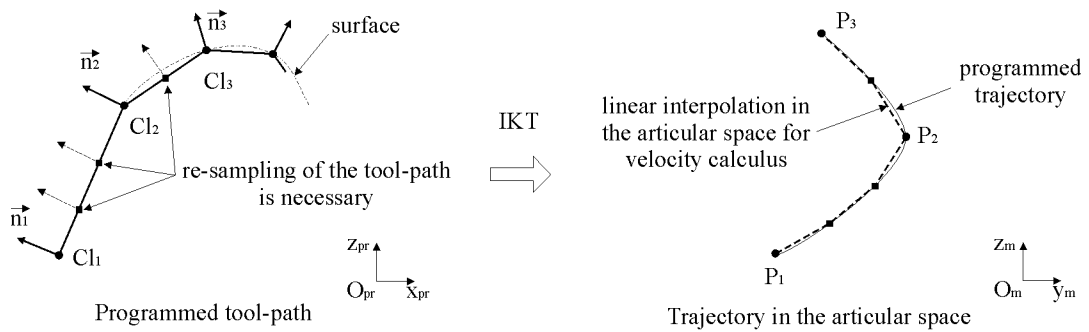


Figure 4. Articular interpolation for velocity calculus.

If a best approximation of the actual velocity is required for the velocity calculus, it can be increased by re-sampling the tool-path [4]. New articular configurations are computed by the IKT and the estimation of the curve length in the articular space is improved.

To summarise, this method enables to coordinate and define axis velocities from the programmed feedrate expressed by the inverse-time.

3.2 Expression of kinematical constraints

Within the context of high speed machining, CNC parameters and components capacities are also critical for the trajectory follow-up. As programmed velocities are very high, axis dynamics and CNC treatment limit kinematical performances. During tests on our system, several constraints are noticed. Three of them seem to be most critical; they are identified and implemented in the model:

- the maximal kinematical capacities;
- the velocity limitation at block transition;
- the velocity limitation due to the cycle time of CNC.

For a serial multi-axis machine, kinematical capacities of each axis are different. They are limited by CNC parameters according their inertia and position into the chain. For each axis, maximum available velocity, acceleration and jerk are respectively $V_{max \text{ axis } j}$, $A_{max \text{ axis } j}$ and $J_{max \text{ axis } j}$. Moreover, during trajectory interpolation, these capacities have to be coordinated to follow-up the trajectory. Hence, given the current position along the trajectory, the feedrate is limited by the less powerful axis. These constraints are integrated in the model by computing the maximum available velocity, acceleration and jerk ($V_{max \text{ inverse-time}}$, $A_{max \text{ inverse-time}}$, $J_{max \text{ inverse-time}}$) for each articular segment (eq.(6) to (8)).

$$V_{max \text{ inverse-time}} = \min \left(\frac{V_{max \text{ axis } j}}{\Delta P_{axis \ j}} \right) \text{ for } \Delta P_{axis \ j} \neq 0 \quad (6)$$

$$A_{max \text{ inverse-time}} = \min \left(\frac{A_{max \text{ axis } j}}{\Delta P_{axis \ j}} \right) \text{ for } \Delta P_{axis \ j} \neq 0 \quad (7)$$

$$J_{max \text{ inverse-time}} = \min \left(\frac{J_{max \text{ axis } j}}{\Delta P_{axis \ j}} \right) \text{ for } \Delta P_{axis \ j} \neq 0 \quad (8)$$

The follow-up also depends on the geometry of the trajectory. In 5-axis machining, discontinuities may appear along the trajectory in the articular space. Tangency and curvature discontinuities involve slowdowns during the follow-up. To prevent velocity drops, the CNC smoothes these discontinuities by inserting arcs of circle on block transition (Figure 5).

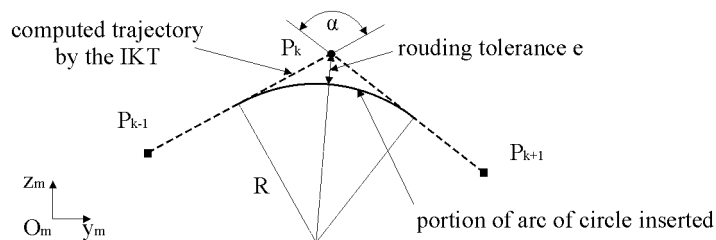


Figure 5. Rounding of tangency discontinuity.

Therefore, velocity is non null but limited by CNC parameters. The post processor models this rounding, taking into account the rounding tolerance value which defines the radius of rounding (eq.(10)) [5][6]. The maximum velocity to cross this rounding can thus be computed with the maximum centripetal acceleration allowed. On the CNC used, the parameter *Curv_Effect_On_Path_Accel* distributes the acceleration capacities between the centripetal acceleration and the tangential acceleration along curved contours (eq. (9)) [6][7]. This velocity limitation is applied at block transition considering that rounding length is short enough to consider that the velocity remains constant on the arc of circle.

$$A_{centripetal\ allowed} = A_{max} \times (1 - Curv_Effect_On_Path_Accel) \quad (9)$$

$$V_{max} = \sqrt{R \times A_{centripetal\ allowed}} \quad \text{with} \quad R = e \times \left(\frac{\sin\left(\frac{\alpha}{2}\right)}{1 - \sin\left(\frac{\alpha}{2}\right)} \right) \quad (10)$$

The last constraint implemented in the post-processor is the maximum velocity relative to the CNC time cycle. As feedrates are high, cycle times have to be short enough to compute new instructions before the previous ones are reached. Otherwise, the axes stop and wait for the next cycle. Indeed, for short segments, the interpolator cycle time can be longer than the duration of axis displacements. For example, if the feedrate is equal to 20m/min and the interpolator cycle time is 3ms, for all segments with a length less than 1mm axes stop at the end of segment by lack of information. To avoid this problem, the maximum axis velocity is limited in order to respect cycle times (eq. (11)).

$$V_{max\ time\ inverse} \leq \frac{1}{interpolator\ cycle\ time} \quad (11)$$

Constraints presented above are expressed by the inverse-time in the post-processor. So, all trajectory long, maximal velocity, acceleration and jerk limit the kinematical performance of each axis. At this stage, the model generates kinematical profiles taking into account these limitations.

3.3 Generation of the axis kinematical profiles

Due to HSM, structures of machine-tools are more and more solicited. High axis velocities and high curvatures require high accelerations. Trajectory discontinuities in the articular space create jerking. Such machine behaviour is harmful for the machine and the part quality. Consequently, jerk limited velocity profile is used to drive axes. This way of piloting reduces jerking and smoother transitions are achieved [8][9]. Actual velocities generated by the CNC are modelled by jerk limited velocity profiles. For each segment of the articular trajectory, a inverse-time feedrate profile is computed from kinematical constraints, initial and final conditions. Finally, axis velocities are reconstructed by multiplying the inverse-time profile by the axis displacements (Figure 6).

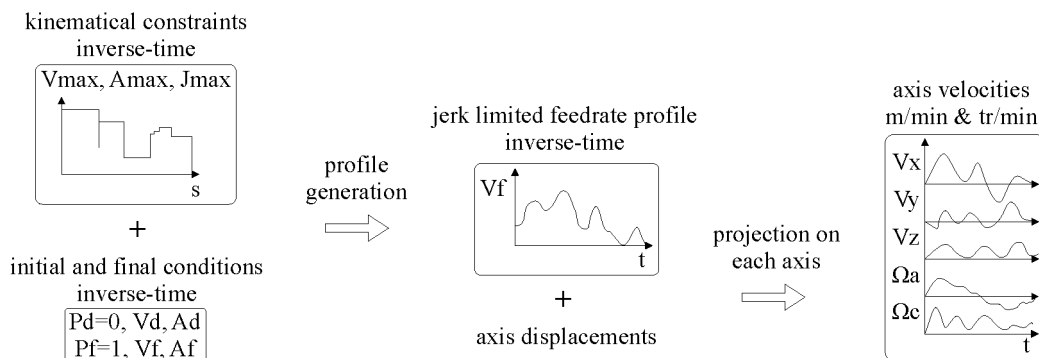


Figure 6. Generation method of axis velocity profiles.

Henceforth, CNC has special functions to improve the follow-up in HSM. As one of them, the dynamic anticipation, is of major influence for high velocities, it is modelled to fit the interpolator treatment. This function, called “look ahead” is used to prevent overshoots. For example, if the anticipation is realised on 10 blocks; in order to generate the profile on block N , the geometry of the trajectory until block $N+10$ is taken into

account. Hence, the velocity deceleration is delayed and overshoots are avoided. The use of such function enables to reach higher velocities (Figure 7).

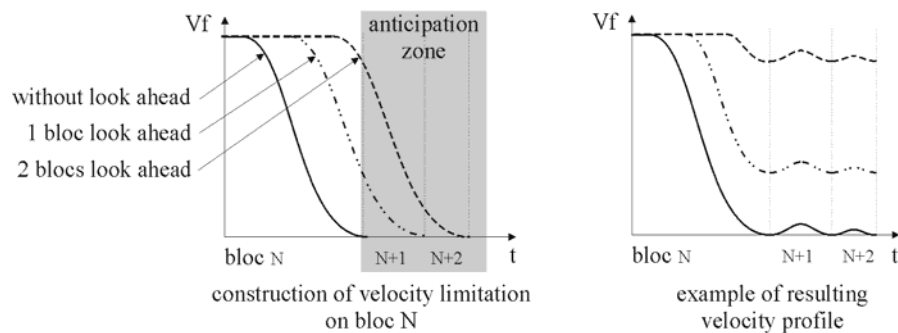


Figure 7. Influence of look ahead function on velocity profile.

4 Application

To illustrate our approach, the machining behaviour on a blending radius of 5 mm is studied (Figure 8). The programmed machining strategy is parallel to plane with toroidal mill ($D=10\text{mm}$, $R_c=1\text{mm}$). Chordal deviation (machining tolerance) is set to 0.01mm ; tool inclination is set to 5° and feedrate to 5 m/min . 5-axis machining is carried out on the Mikron milling centre. The programming frame is oriented on the rotate table such as only the YZA axes are solicited.

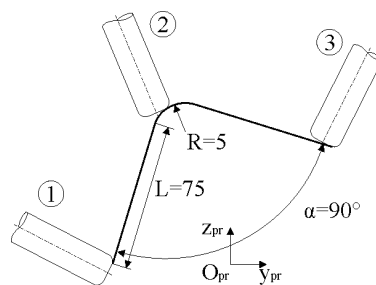


Figure 8. Machining of the blending radius.

Results of predictive model are presented above. First, the IKT is performed. Figure 9 shows the computed axis configurations corresponding to the programmed Cl points.

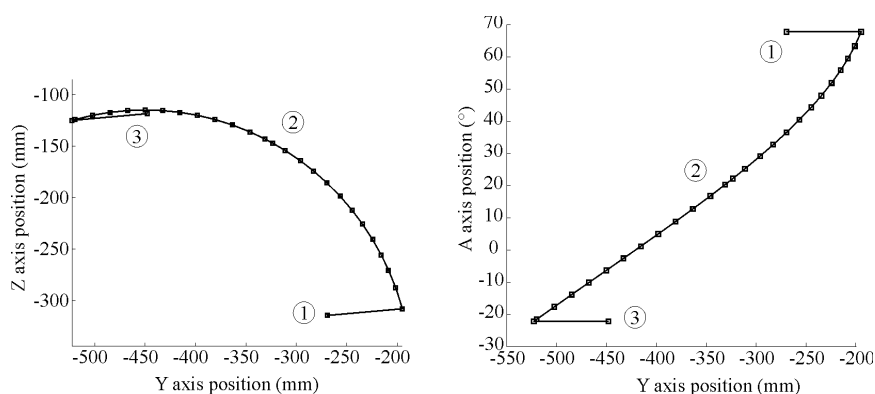


Figure 9. Articular configurations for Cl points.

Given the articular configurations, axis capacities and CNC parameters velocity limitations are predicted along the trajectory (Figure 10). They are expressed in the inverse-time method because profile generation is not yet done.

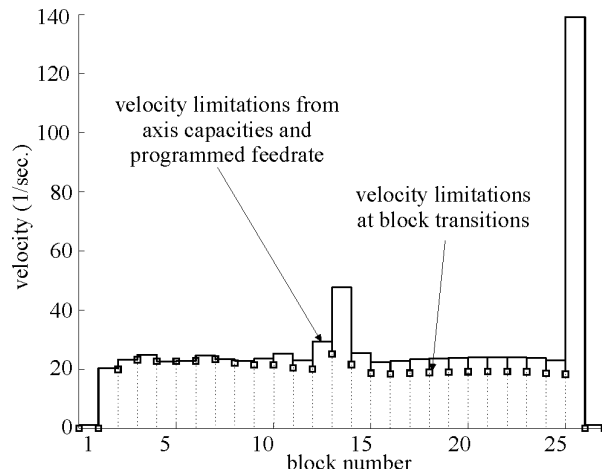


Figure 10. Inverse-time velocity limitations for each block.

The final stage consists in computing the jerk velocity profile taking into account previous constraints. Then, for each segment, the inverse-time velocity is multiplied by axis displacements to obtain each axis velocity. Figure 11 compares the predicted velocities to records done in real time through the CNC during machining.

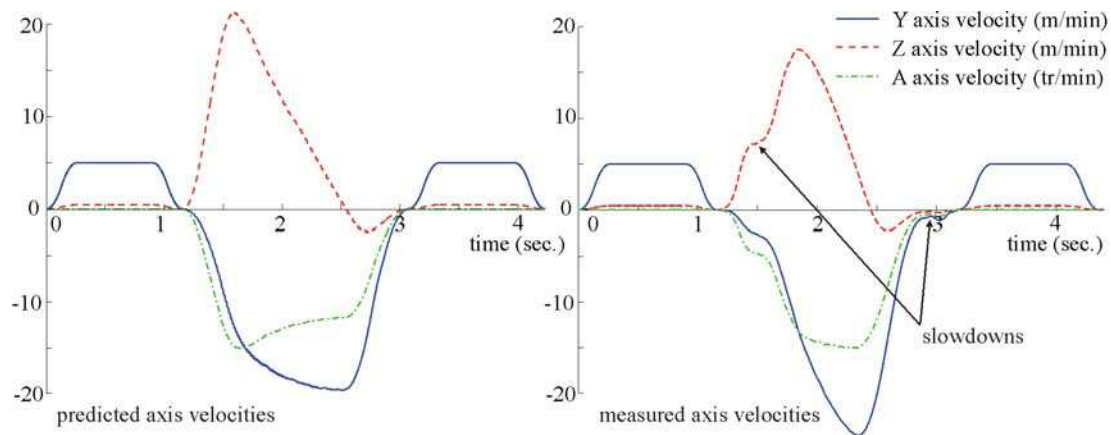


Figure 11. Comparison between predicted and measured velocities.

For the first and the last blocks, where tool axis orientation does not change, predicted axis velocities are equal to the measured one. Differences appear when rotational axis moves. The CNC treatment seems to change when commuting from two axis interpolation (YZ) to three axis interpolation (YZA). Indeed, on block 2, 3, 25 and 26 ($t \approx 1.5$ and 3 sec.), initial and final accelerations are set to zero; slowdowns appear, whereas, between blocks 4 and 24, profile is optimal. This constraint limits axes to reach higher feedrate. So, predicted axis behaviours are locally more dynamic than the actual ones. These treatments modifications by the CNC are difficulties in predicting machining behaviour.

5 Evaluation of cutting conditions

To evaluate the cutting conditions, actual feedrate of the tool relatively to the surface is rebuilt from each axis velocity. Cutting conditions have to be expressed on the Cutter contact point (Cc point). Equation (12) expresses the tool feedrate in the machining direction. It can be approximated by the velocity on Cl point as Cc and Cl points are close enough and rotational axis velocities are low compared to the translational ones (eq. (13) and (14)).

$$\vec{V}_{C_c, tool/surface} = \vec{V}_{C_c, tool/machine} - \vec{V}_{C_c, surface/machine} \quad (12)$$

$$\vec{V}_{C_c, tool/machine} = \vec{V}_{Cl, tool/machine} \quad (13)$$

$$\vec{V}_{C_c, surface/machine} = \vec{V}_{Cl, surface/machine} + \overline{CcCl} \wedge \vec{\Omega}_{surface/machine} \approx \vec{V}_{Cl, surface/machine} \quad (14)$$

Thus, combining Cl positions, axes velocities and machining parameters, cutting conditions are evaluated along the tool-path. Figure 12 shows that programmed velocity is not respected on the bending radius. Indeed, the feedrate reaches 5 m/min along first and last blocks since they are long enough. When the tool axis orientation varies, discontinuities created in the articular space make the feedrate drop to 0 m/min. Along the radius, the maximal reached velocity is about 1 m/min.

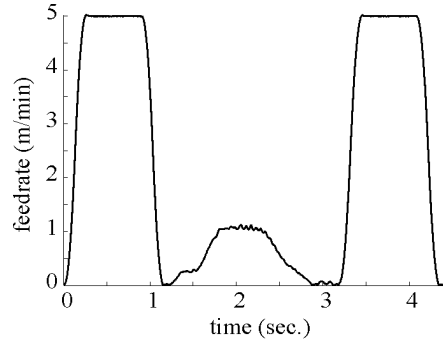


Figure 12. Actual feedrate along the tool-path.

Given the CAXYZ architecture, the actual feedrate depends on axes capacities, chordal deviation, rounding tolerance.... It depends also on the position and orientation of the part in the machine frame. According to the distance between part and rotational axes, variations in tool axis orientation may induce large displacements. These displacements cause high solicitations of translational axes.

To summarize, the reconstruction on the relative feedrate tool-surface gives a criterion to qualify cutting conditions, and so, the quality of the machined part.

6 Conclusion

Multi-axis machining and high velocities create significant solicitations on machine-tool axes, reducing the machined part quality. This is due for the main part to the kinematical performances of the couple CNC-machine-tool. In the paper we have proposed a predictive model of the kinematical behaviour during 5-axis machining to highlight differences between the programmed tool-path and the actual execution of the tool trajectory. For this purpose, a post-processor is developed to approach the Inverse Kinematical Transformation done in real time by the CNC. Limits of the CNC and the machine-tool are exposed. Therefore, axis capacities and CNC special functions are modelled. The inverse-time method used to coordinate axes and to generate jerk limited kinematical profiles constitutes the originality of this model. It allows interpolating axes, whatever their number and their movement (translation or rotation). The results of the predictive model are compared with tests carried out on an industrial milling centre. The global behaviour is similar showing this efficiency of our model. Nevertheless, small differences can be noticed when machining short length segments and for transition between 2 and 3-axis interpolation. Finally, the model builds the relative velocity tool-surface highlighting trajectory portions for which cutting conditions are not respected. Works in progress will now take advantage of our predictive model to find an optimal

machining strategy, computing the tool position and the tool axis orientation, to respect the programmed feedrate as well as possible.

Acknowledgments

This work was carried out within the context of the working group Manufacturing 21 which gathers 11 French research laboratories. The topics approached are: the modelling of the manufacturing process, the virtual machining and the emerging of new manufacturing methods.

References

- [1] A. AFFOUARD, E. DUC, C. LARTIGUE, J.-M. LANGERON, P. BOURDET. *Avoiding 5-axis singularities using tool-path deformation*, International Journal of Machine Tools & Manufacture, N°44, 2004, pp. 415-425.
- [2] Y.H. JUNG, D.W. LEE, J.S. KIM, H.S. MOK. *NC post-processor for 5-axis milling machine of table-rotating/tilting type*, Journal of Materials Processing Technology, N°130-131, 2002, pp.641-646.
- [3] M. MUNLIN, S.S. MAKHANOV, E.L.J. BOHEZ. *Optimization of rotations of a five-axis milling machine near stationary points*, Computer-Aided Design, N°36, 2004, pp. 1117-1128.
- [4] D. VOUILLON, G. POULACHON, L. GASNE. *De la modélisation à l'usinage 5 axes d'un impeller*, Assises MO & UGV, Clermont Ferrand, 2004, pp. 141-150.
- [5] A. DUGAS. *Simulation d'usinage de formes complexes*, PhD Thesis, Ecole Centrale Nantes, December, 2002.
- [6] V. PATELOUP. *Amélioration du comportement cinématique des machines outils UGV. Application au calcul des trajets d'évidement de poches*, PhD Thesis, Blaise Pascal University-Clermont II, July, 2005.
- [7] SINUMERIK 840D/840Di/810D Basic Machine, Description of Functions (FB1), 10.04 Edition, 2001.
- [8] K. ERKORKMAZ, Y. ALTINTAS. *High speed CNC system design. Part I: jerk limited trajectory generation and quintic spline interpolation*, International Journal of Machine Tools & Manufacture, N°41, 2001, pp.1323-1345.
- [9] P.-J. BARRE. *Commande et Entraînement des Machines-Outils à Dynamique Elevée – Formalisme et Applications*, Habilitation à Diriger des Recherches, Université des Sciences et Technologie de Lille, December, 2004.

Appendix: Formulation of the IKT

The architecture of the milling centre Mikron UCP 710 is CAXYZ. Two rotations are applied on the part, and the tool orientation is fixed in the machine frame (Figure 13).

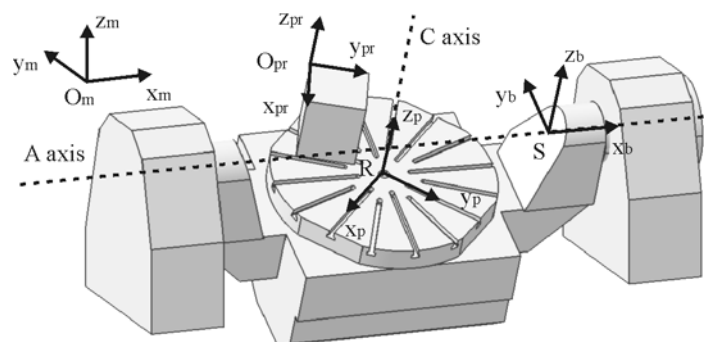


Figure 13. Definition of different frames.

The different frames are:

- the machine frame (O_m, x_m, y_m, z_m) is linked to the structure of the machine-tool. Its axes are parallel to the XYZ axis; the z_m axis of this frame is parallel to the axis of the tool and is pointing to the tool tip;

- the tilt frame (S, x_b, y_b, z_b) is linked to the tilt table: x_b is parallel to x_m , S is located on the A axis and is given by equation (15);

$$\overrightarrow{OmS} = m_x \cdot \overrightarrow{x_m} + m_y \cdot \overrightarrow{y_m} + m_z \cdot \overrightarrow{z_m} \quad (15)$$

- the table frame (R, x_p, y_p, z_p) is linked to the rotary table: z_p is parallel to z_b , R is defined as the intersection between the C axis and the upper face of the table (eq. (16));

$$\overrightarrow{SR} = b_y \cdot \overrightarrow{y_b} + b_z \cdot \overrightarrow{z_b} \quad (16)$$

- the programming frame ($O_{pr}, x_{pr}, y_{pr}, z_{pr}$) is linked to the part: it represents the frame used for CAM computation. Its origin O_{pr} is given by equation (17).

$$\overrightarrow{ROpr} = p_x \cdot \overrightarrow{x_p} + p_y \cdot \overrightarrow{y_p} + p_z \cdot \overrightarrow{z_p} \quad (17)$$

The parameters m_x, m_y, m_z, b_y and b_z are fixed values, identified on the machine-tool. The position of the part on the table define the parameters p_x, p_y and p_z . To switch easily between different frames, we generally define P_{ij} the matrix that converts a vector initially expressed in the frame j into the frame i : $\vec{V}_{frame i} = P_{ij} \cdot \vec{V}_{frame j}$ (eq. (18)).

$$P_{mb} = \begin{bmatrix} 1 & 0 & 0 & m_x \\ 0 & \cos(A) & \sin(A) & m_y \\ 0 & -\sin(A) & \cos(A) & m_z \\ 0 & 0 & 0 & 1 \end{bmatrix} P_{bp} = \begin{bmatrix} \cos(C) & \sin(C) & 0 & 0 \\ -\sin(C) & \cos(C) & 0 & b_y \\ 0 & 0 & 1 & b_z \\ 0 & 0 & 0 & 1 \end{bmatrix} P_{ppr} = \begin{bmatrix} a & d & g & p_x \\ b & e & h & p_y \\ c & f & i & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (18)$$

A and C are the angular values to command rotational axis; The parameters $a, b, c, d, e, f, g, h, i$ define the orientation of the part on the table.

Equation (19) expresses that the tool orientation is fixed in the machine frame; the tool orientation on the surface is given by the two rotational axes on the part.

$$\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}_{(x_m, y_m, z_m)} = P_{mb} \times P_{bp} \times P_{ppr} \times \begin{bmatrix} i \\ j \\ k \\ 0 \end{bmatrix}_{(x_{pr}, y_{pr}, z_{pr})} \quad (19)$$

Then, once the values of angles A and C are computed, the X, Y and Z axis correct the displacements induced on the part and move the tool to reach the CL point. The commands to apply to axis of translation are given by equation (20).

$$\begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix}_{(O_m, x_m, y_m, z_m)} = P_{mb} \times P_{bp} \times P_{ppr} \times \begin{bmatrix} X_{pr} \\ Y_{pr} \\ Z_{pr} \\ 1 \end{bmatrix}_{(O_{pr}, x_{pr}, y_{pr}, z_{pr})} \quad (20)$$